Non-intrusive model reduction of shift-equivariant systems via data-driven projection and reduced dynamics

Yu Shuai

October 29, 2025

2025-10-29

Non-intrusive model reduction of shift-equivariant systems via data-driven projection and reduced dynamics

Non-intrusive model reduction of shift-equivariant systems via data-driven projection and reduced dynamics

Yu Shuai

October 29, 2025

Contents

2025-10-29

Non-intrusive model reduction of shift-equivariant systems via data-driven projection and reduced dynamics

└─Contents

Contents

1 Before we start: Photos from the West

2 Method

3 Results

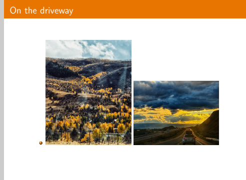
Contents

Contents
Before we start: Photos from the West
Photos
Method
Results

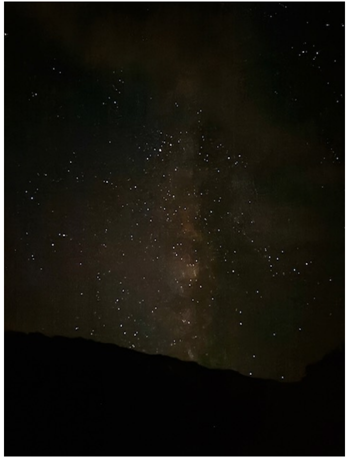
On the driveway

2025-10-29

Non-intrusive model reduction of shift-equivariant systems via data-driven projection and reduced dynamics
└Before we start: Photos from the West
  └Photos



The Arches National Park

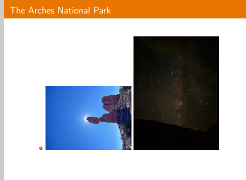


2025-10-29

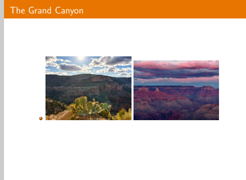
Non-intrusive model reduction of shift-equivariant systems via data-driven projection and reduced dynamics

└Before we start: Photos from the West

└Photos



The Grand Canyon

Contents

Contents
Before we start: Photos from the West
Method
Symmetry-reduced non-intrusive trajectory-based ROM (SR-NiTROM)
Results

Formulation of the SR-NiTROM

- Consider a shift-equivariant FOM with traveling solutions:

$$u_t = f(u), \qquad u(x,t) = \hat{u}(x - c(t), t). \tag{1}$$

Formulation of the SR-NiTROM
Consider a shift-equivariant FOM with traveling solutions:
$u_t = f(u), \quad u(x,t) = \hat{u}(x - c(t), t).$ (1)

Formulation of the SR-NiTROM

- Consider a shift-equivariant FOM with traveling solutions:

$$u_t = f(u), \qquad u(x,t) = \hat{u}(x - c(t), t). \tag{1}$$

- Suppose we have a collection of training snapshots $\{u(t_m)\}_{m=0}^{N_t-1}$, $u(t) \in \mathbb{R}^N$.

Formulation of the SR-NiTROM
- Consider a shift-equivariant FOM with traveling solutions:
$$u_t = f(u), \qquad u(x,t) = \hat{u}(x - c(t), t). \tag{1}$$
- Suppose we have a collection of training snapshots $\{u(t_m)\}_{m=0}^{N_t-1}$, $u(t) \in \mathbb{R}^N$.

Formulation of the SR-NiTROM

- Consider a shift-equivariant FOM with traveling solutions:

$$u_t = f(u), \qquad u(x,t) = \hat{u}(x - c(t), t). \tag{1}$$

- Suppose we have a collection of training snapshots $\{u(t_m)\}_{m=0}^{N_t-1}$, $u(t) \in \mathbb{R}^N$.
- We seek to find $\Phi, \Psi \in \mathbb{R}^{n\times r}$, such that $\Phi(\Psi^\top \Phi)^{-1}\Psi^\top$ is a projection.

Formulation of the SR-NiTROM
Consider a shift-equivariant FOM with traveling solutions:
$u_t = f(u), \qquad u(x,t) = \hat{u}(x - c(t), t).$ (1)
Suppose we have a collection of training snapshots $\{u(t_m)\}_{m=0}^{N_t-1}$, $u(t) \in \mathbb{R}^N$.
We seek to find $\Phi, \Psi \in \mathbb{R}^{n\times r}$, such that $\Phi(\Psi^\top \Phi)^{-1}\Psi^\top$ is a projection.

Formulation of the SR-NiTROM

- Consider a shift-equivariant FOM with traveling solutions:

$$u_t = f(u), \qquad u(x,t) = \hat{u}(x - c(t), t). \tag{1}$$

- Suppose we have a collection of training snapshots $\{u(t_m)\}_{m=0}^{N_t-1}$, $u(t) \in \mathbb{R}^N$.
- We seek to find $\Phi, \Psi \in \mathbb{R}^{n\times r}$, such that $\Phi(\Psi^\top \Phi)^{-1}\Psi^\top$ is a projection.
- From this projection operator, we can encode the FOM state with a low-dim representation $a(t) \in \mathbb{R}^r$:

$$\begin{aligned} a &= \Psi^\top u \\ \hat{u}_r &= \Phi(\Psi^\top \Phi)^{-1} a. \end{aligned} \tag{2}$$

Formulation of the SR-NiTROM
- Consider a shift-equivariant FOM with traveling solutions:
$$u_t = f(u), \qquad u(x,t) = \hat{u}(x - c(t), t). \tag{1}$$
- Suppose we have a collection of training snapshots $\{u(t_m)\}_{m=0}^{N_t-1}$, $u(t) \in \mathbb{R}^N$.
- We seek to find $\Phi, \Psi \in \mathbb{R}^{n\times r}$, such that $\Phi(\Psi^\top \Phi)^{-1}\Psi^\top$ is a projection.
- From this projection operator, we can encode the FOM state with a low-dim representation $a(t) \in \mathbb{R}^r$:
$$\begin{aligned} a &= \Psi^\top u \\ \hat{u}_r &= \Phi(\Psi^\top \Phi)^{-1} a. \end{aligned} \tag{2}$$

- Consider a shift-equivariant FOM with traveling solutions:

$$u_t = f(u), \qquad u(x,t) = \hat{u}(x - c(t), t). \tag{1}$$

- Suppose we have a collection of training snapshots $\{u(t_m)\}_{m=0}^{N_t-1}$, $u(t) \in \mathbb{R}^N$.
- We seek to find $\Phi, \Psi \in \mathbb{R}^{n\times r}$, such that $\Phi(\Psi^\top \Phi)^{-1}\Psi^\top$ is a projection.
- From this projection operator, we can encode the FOM state with a low-dim representation $a(t) \in \mathbb{R}^r$:

$$\begin{aligned} a &= \Psi^\top u \\ \hat{u}_r &= \Phi(\Psi^\top\Phi)^{-1} a. \end{aligned} \tag{2}$$

- The ROM dynamics is given in a symmetry-reduced form:

$$\dot{a}_i = A_{ij}a_j + B_{ijk}a_j a_k + \dot{c}M_{ij}a_j \tag{3a}$$

$$\dot{c} = -\frac{p_i a_i + Q_{ij}a_i a_j}{s_i a_i} \tag{3b}$$

$$M = \Psi^\top \partial_x \Phi(\Psi^\top\Phi)^{-1}, \quad s = \langle \partial_x \Phi(\Psi^\top\Phi)^{-1}, \partial_x u_0 \rangle \tag{3c}$$

Formulation of the SR-NiTROM
Consider a shift-equivariant FOM with traveling solutions:
Suppose we have a collection of training snapshots
We seek to find $\Phi, \Psi \in \mathbb{R}^{n\times r}$, such that $\Phi(\Psi^\top \Phi)^{-1}\Psi^\top$ is a projection.
From this projection operator, we can encode the FOM state with a low-dim representation $a(t) \in \mathbb{R}^r$:
The ROM dynamics is given in a symmetry-reduced form:

The optimization problem of SR-NiTROM

- The trajectory-based objective function:

$$J = \sum_{m=0}^{N_t-1} \|\hat{u}_r(t_m) - \hat{u}(t_m)\|^2 + \beta(c_r(t_m) - c(t_m))^2. \tag{4}$$

The optimization problem of SR-NiTROM
The trajectory-based objective function:
$$J = \sum_{m=0}^{N_t-1} \|\hat{u}_r(t_m) - \hat{u}(t_m)\|^2 + \beta(c_r(t_m) - c(t_m))^2. \tag{4}$$

The optimization problem of SR-NiTROM

- The trajectory-based objective function:

$$J = \sum_{m=0}^{N_t-1} \|\hat{u}_r(t_m) - \hat{u}(t_m)\|^2 + \beta(c_r(t_m) - c(t_m))^2. \tag{4}$$

  - $\beta = \gamma \sum_m \|\hat{u}(t_m)\|^2 / \sum_m (c(t_0) - c(t_m))^2$: relative weights.

The optimization problem of SR-NiTROM
- The trajectory-based objective function:
$$J = \sum_{m=0}^{N_t-1} \|\hat{u}_r(t_m) - \hat{u}(t_m)\|^2 + \beta(c_r(t_m) - c(t_m))^2. \tag{4}$$
- $\beta = \gamma \sum_m \|\hat{u}(t_m)\|^2 / \sum_m (c(t_0) - c(t_m))^2$: relative weights.

The optimization problem of SR-NiTROM

- The trajectory-based objective function:

$$J = \sum_{m=0}^{N_t-1} \|\hat{u}_r(t_m) - \hat{u}(t_m)\|^2 + \beta(c_r(t_m) - c(t_m))^2. \tag{4}$$

  - $\beta = \gamma \sum_m \|\hat{u}(t_m)\|^2 / \sum_m (c(t_0) - c(t_m))^2$: relative weights.
  - γ: hyperparameter.

The optimization problem of SR-NiTROM

- The trajectory-based objective function:

$$J = \sum_{m=0}^{N_t-1} \|\hat{u}_r(t_m) - \hat{u}(t_m)\|^2 + \beta(c_r(t_m) - c(t_m))^2. \tag{4}$$

  - $\beta = \gamma \sum_m \|\hat{u}(t_m)\|^2 / \sum_m (c(t_0) - c(t_m))^2$: relative weights.
  - γ: hyperparameter.

The optimization problem of SR-NiTROM

- The trajectory-based objective function:

$$J = \sum_{m=0}^{N_t - 1} \|\hat{u}_r(t_m) - \hat{u}(t_m)\|^2 + \beta(c_r(t_m) - c(t_m))^2. \tag{4}$$

  - $\beta = \gamma \sum_m \|\hat{u}(t_m)\|^2 / \sum_m (c(t_0) - c(t_m))^2$: relative weights.
  - γ: hyperparameter.
- The unconstrained Lagrangian with multipliers:

$$L = \sum_{m=0}^{N_t-1} \Bigg(\|\hat{u}_r(t_m) - \hat{u}(t_m)\|^2 + \beta(c_r(t_m) - c(t_m))$$

$$+ \int_{t_0}^{t_m} \lambda_m^\top (\dot{a} - Aa - B(a, a) - \dot{c}Ma)\mathrm{d}t \tag{5}$$

$$+ \int_{t_0}^{t_m} \mu_m (\dot{c} + \frac{p_i a_i + Q_{ij} a_i a_j}{s_i a_i})\mathrm{d}t \tag{6}$$

$$+ \lambda_m(t_0)(a(t_0) - \Psi^\top \hat{u}(t_0)) \Bigg), \quad \lambda_m \in \mathbb{R}^r, \mu_m \in \mathbb{R}. \tag{7}$$

The optimization problem of SR-NiTROM
The trajectory-based objective function:
$J = \sum_{m=0}^{N_t - 1} \|\hat{u}_r(t_m) - \hat{u}(t_m)\|^2 + \beta(c_r(t_m) - c(t_m))^2.$ (4)
$\beta = \gamma \sum_m \|\hat{u}(t_m)\|^2 / \sum_m (c(t_0) - c(t_m))^2$: relative weights.
γ: hyperparameter.
The unconstrained Lagrangian with multipliers:
$L = \sum_{m=0}^{N_t-1} \Big(\|\hat{u}_r(t_m) - \hat{u}(t_m)\|^2 + \beta(c_r(t_m) - c(t_m))$
$+ \int_{t_0}^{t_m} \lambda_m^\top (\dot{a} - Aa - B(a, a) - \dot{c}Ma)\mathrm{d}t$ (5)
$+ \int_{t_0}^{t_m} \mu_m (\dot{c} + \frac{p_i a_i + Q_{ij} a_i a_j}{s_i a_i})\mathrm{d}t$ (6)
$+ \lambda_m(t_0)(a(t_0) - \Psi^\top \hat{u}(t_0)) \Big), \quad \lambda_m \in \mathbb{R}^r, \mu_m \in \mathbb{R}.$ (7)

Contents

Contents

Before we start: Photos from the West

Method

Results

- Reconstruction of a single training KSE trajectory with dim-4 ROMs: SR-NiTROM vs SR-Galerkin
- Reconstruction of multiple training KSE trajectories with dim-8 ROMs
- Reconstruction of multiple testing KSE trajectories with dim-8 ROMs
- Reconstruction of multiple training KSE trajectories with dim-16 ROMs
- Ongoing: Reconstruction of channel-flow solutions to the 3D linearized Navier-Stokes equations

Numerical details

- FOM: Kuramoto-Sivashinsky equation

$$u_t = -uu_x - u_{xx} - \nu u_{xxxx}, \quad x \in [0, 2\pi]. \tag{8}$$

Numerical details
FOM: Kuramoto-Sivashinsky equation
$$u_t = -uu_x - u_{xx} - \nu u_{xxxx}, \quad x \in [0, 2\pi]. \tag{8}$$

Numerical details

- FOM: Kuramoto-Sivashinsky equation

$$u_t = -u u_x - u_{xx} - \nu u_{xxxx}, \quad x \in [0, 2\pi]. \tag{8}$$

  - $\nu = 4/87$ for traveling-wave patterns.

Numerical details

- FOM: Kuramoto-Sivashinsky equation

$$u_t = -u u_x - u_{xx} - \nu u_{xxxx}, \quad x \in [0, 2\pi]. \tag{8}$$

  - $\nu = 4/87$ for traveling-wave patterns.

Numerical details

- FOM: Kuramoto-Sivashinsky equation

$$u_t = -u u_x - u_{xx} - \nu u_{xxxx}, \quad x \in [0, 2\pi]. \tag{8}$$

  - $\nu = 4/87$ for traveling-wave patterns.
  - Periodic BCs, $N = 40$ Fourier modes, $\Delta t = 10^{-3}$.

Numerical details
FOM: Kuramoto-Sivashinsky equation
$$u_t = -u u_x - u_{xx} - \nu u_{xxxx}, \quad x \in [0, 2\pi]. \tag{8}$$
$\nu = 4/87$ for traveling-wave patterns.
Periodic BCs, $N = 40$ Fourier modes, $\Delta t = 10^{-3}$.

Numerical details

- FOM: Kuramoto-Sivashinsky equation

$$u_t = -uu_x - u_{xx} - \nu u_{xxxx}, \quad x \in [0, 2\pi]. \tag{8}$$

  - $\nu = 4/87$ for traveling-wave patterns.
  - Periodic BCs, $N = 40$ Fourier modes, $\Delta t = 10^{-3}$.
  - Sample interval: 10 timesteps between 2 adjacent snapshots.

Numerical details
FOM: Kuramoto-Sivashinsky equation
$$u_t = -uu_x - u_{xx} - \nu u_{xxxx}, \quad x \in [0, 2\pi]. \tag{8}$$
$\nu = 4/87$ for traveling-wave patterns.
Periodic BCs, $N = 40$ Fourier modes, $\Delta t = 10^{-3}$.
Sample interval: 10 timesteps between 2 adjacent snapshots.

Numerical details

- FOM: Kuramoto-Sivashinsky equation

$$u_t = -uu_x - u_{xx} - \nu u_{xxxx}, \quad x \in [0, 2\pi]. \tag{8}$$

  - $\nu = 4/87$ for traveling-wave patterns.
  - Periodic BCs, $N = 40$ Fourier modes, $\Delta t = 10^{-3}$.
  - Sample interval: 10 timesteps between 2 adjacent snapshots.
- Optimization of the SR-NiTROM:

Numerical details
- FOM: Kuramoto-Sivashinsky equation
$$u_t = -uu_x - u_{xx} - \nu u_{xxxx}, \quad x \in [0, 2\pi]. \tag{8}$$
  - $\nu = 4/87$ for traveling-wave patterns.
  - Periodic BCs, $N = 40$ Fourier modes, $\Delta t = 10^{-3}$.
  - Sample interval: 10 timesteps between 2 adjacent snapshots.
- Optimization of the SR-NiTROM:

Numerical details

- FOM: Kuramoto-Sivashinsky equation

$$u_t = -uu_x - u_{xx} - \nu u_{xxxx}, \quad x \in [0, 2\pi]. \tag{8}$$

  - $\nu = 4/87$ for traveling-wave patterns.
  - Periodic BCs, $N = 40$ Fourier modes, $\Delta t = 10^{-3}$.
  - Sample interval: 10 timesteps between 2 adjacent snapshots.
- Optimization of the SR-NiTROM:
  - Alternating training: 5 iterations for the tensors fixing the bases, then 5 iterations for the bases fixing the tensors. Switch 20 times.

Numerical details
- FOM: Kuramoto-Sivashinsky equation
$$u_t = -uu_x - u_{xx} - \nu u_{xxxx}, \quad x \in [0, 2\pi]. \tag{8}$$
  - $\nu = 4/87$ for traveling-wave patterns.
  - Periodic BCs, $N = 40$ Fourier modes, $\Delta t = 10^{-3}$.
  - Sample interval: 10 timesteps between 2 adjacent snapshots.
- Optimization of the SR-NiTROM:
  - Alternating training: 5 iterations for the tensors fixing the bases, then 5 iterations for the bases fixing the tensors. Switch 20 times.

Numerical details

- FOM: Kuramoto-Sivashinsky equation

$$u_t = -uu_x - u_{xx} - \nu u_{xxxx}, \quad x \in [0, 2\pi]. \tag{8}$$

  - $\nu = 4/87$ for traveling-wave patterns.
  - Periodic BCs, $N = 40$ Fourier modes, $\Delta t = 10^{-3}$.
  - Sample interval: 10 timesteps between 2 adjacent snapshots.
- Optimization of the SR-NiTROM:
  - Alternating training: 5 iterations for the tensors fixing the bases, then 5 iterations for the bases fixing the tensors. Switch 20 times.
  - Conjugate-gradient optimizer with line search method.

Numerical details

- FOM: Kuramoto-Sivashinsky equation

$$u_t = -uu_x - u_{xx} - \nu u_{xxxx}, \quad x \in [0, 2\pi]. \tag{8}$$

  - $\nu = 4/87$ for traveling-wave patterns.
  - Periodic BCs, $N = 40$ Fourier modes, $\Delta t = 10^{-3}$.
  - Sample interval: 10 timesteps between 2 adjacent snapshots.
- Optimization of the SR-NiTROM:
  - Alternating training: 5 iterations for the tensors fixing the bases, then 5 iterations for the bases fixing the tensors. Switch 20 times.
  - Conjugate-gradient optimizer with line search method.

- FOM: Kuramoto-Sivashinsky equation

$$u_t = -uu_x - u_{xx} - \nu u_{xxxx}, \quad x \in [0, 2\pi]. \tag{8}$$

  - $\nu = 4/87$ for traveling-wave patterns.
  - Periodic BCs, $N = 40$ Fourier modes, $\Delta t = 10^{-3}$.
  - Sample interval: 10 timesteps between 2 adjacent snapshots.
- Optimization of the SR-NiTROM:
  - Alternating training: 5 iterations for the tensors fixing the bases, then 5 iterations for the bases fixing the tensors. Switch 20 times.
  - Conjugate-gradient optimizer with line search method.
  - Initial conditions: POD bases (capturing >99.5% energy) + Galerkin-projected tensors. (imitating the training result of the re-projected SR-OpInf ROM)

Numerical details

- FOM: Kuramoto-Sivashinsky equation

$$u_t = -uu_x - u_{xx} - \nu u_{xxxx}, \quad x \in [0, 2\pi]. \tag{8}$$

  - $\nu = 4/87$ for traveling-wave patterns.
  - Periodic BCs, $N = 40$ Fourier modes, $\Delta t = 10^{-3}$.
  - Sample interval: 10 timesteps between 2 adjacent snapshots.
- Optimization of the SR-NiTROM:
  - Alternating training: 5 iterations for the tensors fixing the bases, then 5 iterations for the bases fixing the tensors. Switch 20 times.
  - Conjugate-gradient optimizer with line search method.
  - Initial conditions: POD bases (capturing >99.5% energy) + Galerkin-projected tensors. (imitating the training result of the re-projected SR-OpInf ROM)

Numerical details

- FOM: Kuramoto-Sivashinsky equation

$$u_t = -uu_x - u_{xx} - \nu u_{xxxx}, \quad x \in [0, 2\pi]. \tag{8}$$

  - $\nu = 4/87$ for traveling-wave patterns.
  - Periodic BCs, $N = 40$ Fourier modes, $\Delta t = 10^{-3}$.
  - Sample interval: 10 timesteps between 2 adjacent snapshots.
- Optimization of the SR-NiTROM:
  - Alternating training: 5 iterations for the tensors fixing the bases, then 5 iterations for the bases fixing the tensors. Switch 20 times.
  - Conjugate-gradient optimizer with line search method.
  - Initial conditions: POD bases (capturing >99.5% energy) + Galerkin-projected tensors. (imitating the training result of the re-projected SR-OpInf ROM)
  - If the training is not stable, then train on short trajectories first and then extend in time gradually ("curriculum learning").

Numerical details
- FOM: Kuramoto-Sivashinsky equation
$$u_t = -uu_x - u_{xx} - \nu u_{xxxx}, \quad x \in [0, 2\pi]. \tag{8}$$
  - $\nu = 4/87$ for traveling-wave patterns.
  - Periodic BCs, $N = 40$ Fourier modes, $\Delta t = 10^{-3}$.
  - Sample interval: 10 timesteps between 2 adjacent snapshots.
- Optimization of the SR-NiTROM:
  - Alternating training: 5 iterations for the tensors fixing the bases, then 5 iterations for the bases fixing the tensors. Switch 20 times.
  - Conjugate-gradient optimizer with line search method.
  - Initial conditions: POD bases (capturing >99.5% energy) + Galerkin-projected tensors. (imitating the training result of the re-projected SR-OpInf ROM)
  - If the training is not stable, then train on short trajectories first and then extend in time gradually ("curriculum learning").

Results: single transient trajectory from t = 30 to t = 40

- Relative weight: 10.0, 4-dim ROM

2025-10-29

Non-intrusive model reduction of shift-equivariant systems via data-driven projection and reduced dynamics

└─Results

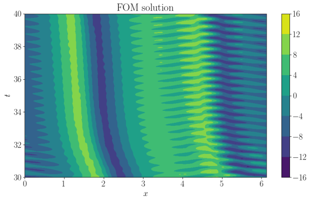
└─Reconstruction of a single training KSE trajectory

Results: single transient trajectory from t = 30 to t = 40

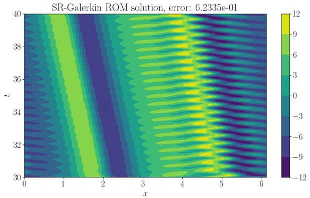
- Relative weight: 10.0, 4-dim ROM

Results: single transient trajectory from t = 30 to t = 40

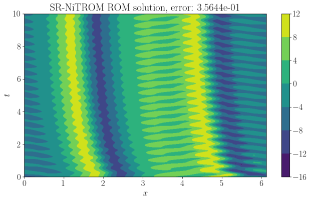
- Relative weight: 10.0, 4-dim ROM
- FOM vs SR-Galerkin vs SR-NiTROM



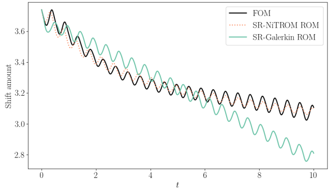
(a) FOM



(b) SR-Galerkin

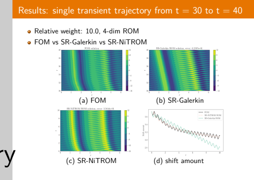


(c) SR-NiTROM



(d) shift amount

Non-intrusive model reduction of shift-equivariant systems via data-driven projection and reduced dynamics
└─Results
└─Reconstruction of a single training KSE trajectory



Results: multiple trajectories from perturbed initial conditions

- 9 trajectories (time length $= 10$), 8-dim ROM

Results: multiple trajectories from perturbed initial conditions

- 9 trajectories (time length $= 10$), 8-dim ROM

Results: multiple trajectories from perturbed initial conditions

- 9 trajectories (time length $= 10$), 8-dim ROM
- Initial conditions: post-transient solution snapshot $+$ perturbations $u(t = 80) + \{0, \sin(x), ..., \sin(4x), \cos(x), ..., \cos(4x)\}$.

Results: multiple trajectories from perturbed initial conditions

- 9 trajectories (time length $= 10$), 8-dim ROM
- Initial conditions: post-transient solution snapshot $+$ perturbations $u(t = 80) + \{0, \sin(x), ..., \sin(4x), \cos(x), ..., \cos(4x)\}$.

Results: multiple trajectories from perturbed initial conditions

- 9 trajectories (time length $= 10$), 8-dim ROM
- Initial conditions: post-transient solution snapshot + perturbations $u(t = 80) + \{0, \sin(x), ..., \sin(4x), \cos(x), ..., \cos(4x)\}$.
- Two types of errors: The raw L_2 error and the fitted L_2 error.

$$\epsilon = \frac{\sum_{m=0}^{N_t-1} \|u(t) - u_r(t)\|_2^2}{\sum_{m=0}^{N_t-1} \|u(t)\|_2^2} \tag{9}$$

$$\epsilon_f = \frac{\sum_{m=0}^{N_t-1} \|\hat{u}(t) - \hat{u}_r(t)\|_2^2}{\sum_{m=0}^{N_t-1} \|\hat{u}(t)\|_2^2} \tag{10}$$

2025-10-29

Non-intrusive model reduction of shift-equivariant systems via data-driven projection and reduced dynamics
└─Results
└─Reconstruction of multiple training KSE trajectories

Results: multiple trajectories from perturbed initial conditions

- 9 trajectories (time length $= 10$), 8-dim ROM
- Initial conditions: post-transient solution snapshot + perturbations $u(t = 80) + \{0, \sin(x), ..., \sin(4x), \cos(x), ..., \cos(4x)\}$.
- Two types of errors: The raw L_2 error and the fitted L_2 error.

$$\epsilon = \frac{\sum_{m=0}^{N_t-1} \|u(t) - u_r(t)\|_2^2}{\sum_{m=0}^{N_t-1} \|u(t)\|_2^2} \tag{9}$$

$$\epsilon_f = \frac{\sum_{m=0}^{N_t-1} \|\hat{u}(t) - \hat{u}_r(t)\|_2^2}{\sum_{m=0}^{N_t-1} \|\hat{u}(t)\|_2^2} \tag{10}$$

Results: multiple trajectories from perturbed initial conditions

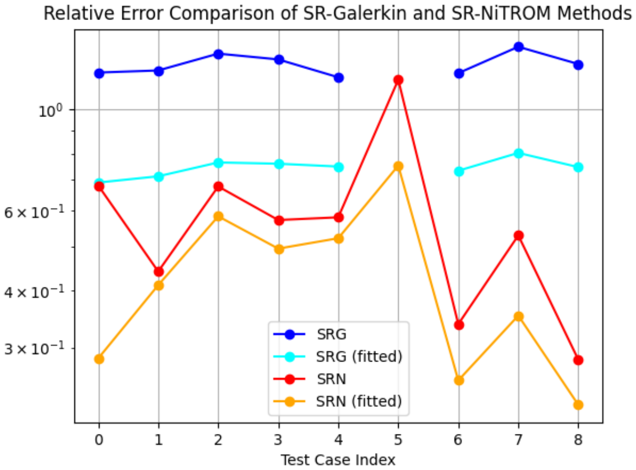


Figure: Comparison of relative reconstruction errors of training set. ROM

Results: multiple trajectories from perturbed initial conditions

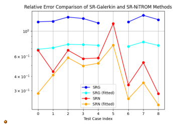
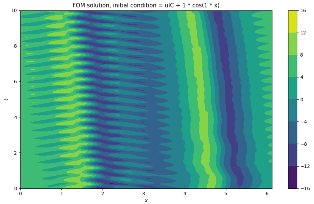


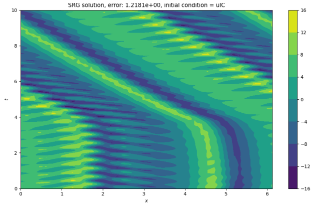
Figure: Comparison of relative reconstruction errors of training set. ROM dimension $r = 8$.

Results: multiple trajectories from perturbed initial conditions

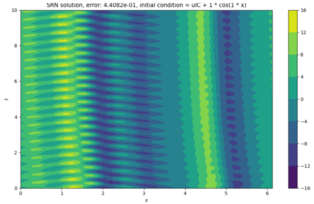
- Trajectory 2:



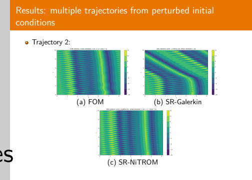
(a) FOM



(b) SR-Galerkin

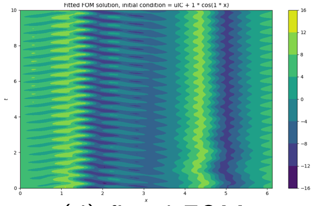


(c) SR-NiTROM

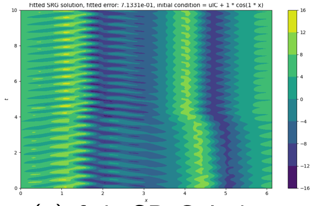


Results: multiple trajectories from perturbed initial conditions

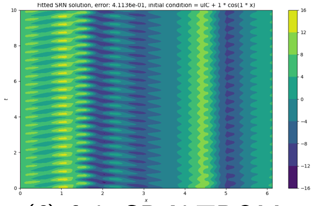
- Trajectory 2:



(d) fitted FOM



(e) ftd. SR-Galerkin

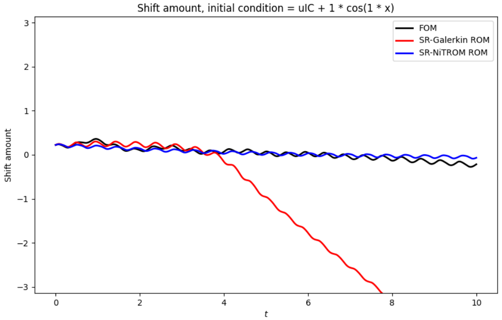


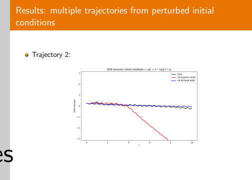
(f) ftd. SR-NiTROM

Results: multiple trajectories from perturbed initial conditions
Trajectory 2:
(d) fitted FOM (e) ftd. SR-Galerkin
(f) ftd. SR-NiTROM

Results: multiple trajectories from perturbed initial conditions

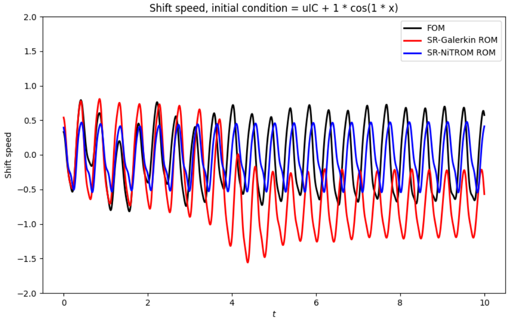
- Trajectory 2:





Results: multiple trajectories from perturbed initial conditions

- Trajectory 2:

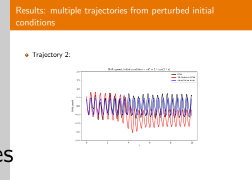


2025-10-29

Non-intrusive model reduction of shift-equivariant systems via data-driven projection and reduced dynamics

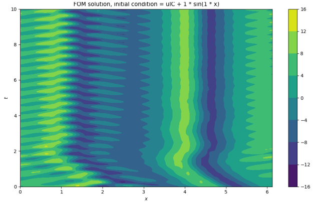
└─Results

└─Reconstruction of multiple training KSE trajectories

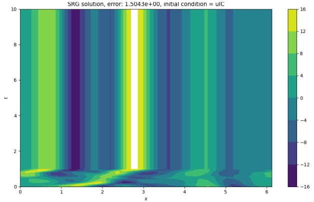


Results: multiple trajectories from perturbed initial conditions

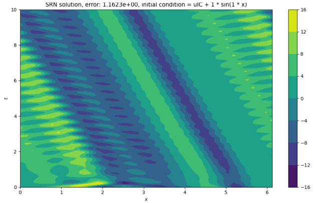
- Trajectory 6 (where the SR-Galerkin ROM solution blows up):



(a) FOM



(b) SR-Galerkin



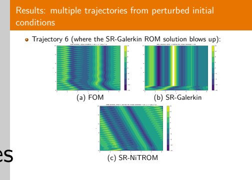
(c) SR-NiTROM

2025-10-29

Non-intrusive model reduction of shift-equivariant systems via data-driven projection and reduced dynamics

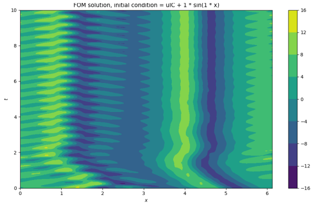
└Results

└Reconstruction of multiple training KSE trajectories

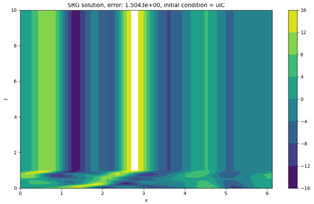


Results: multiple trajectories from perturbed initial conditions

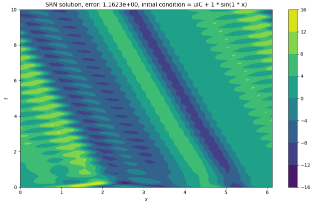
- Trajectory 6 (where the SR-Galerkin ROM solution blows up):



(a) FOM



(b) SR-Galerkin



(c) SR-NiTROM

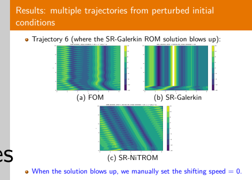
- When the solution blows up, we manually set the shifting speed = 0.

2025-10-29

Non-intrusive model reduction of shift-equivariant systems via data-driven projection and reduced dynamics

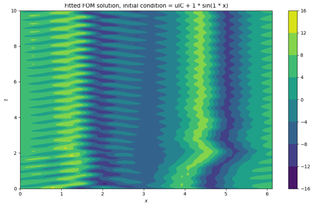
└─Results

└─Reconstruction of multiple training KSE trajectories

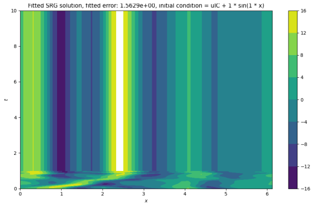


Results: multiple trajectories from perturbed initial conditions

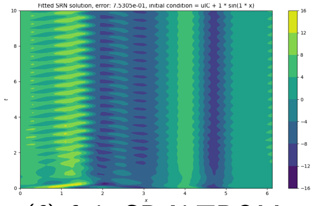
- Trajectory 6 (where the SR-Galerkin ROM solution blows up):



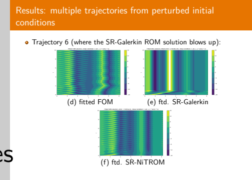
(d) fitted FOM



(e) ftd. SR-Galerkin

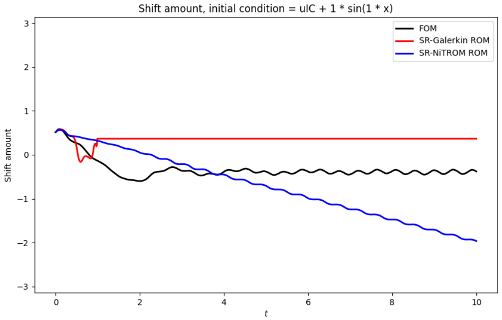


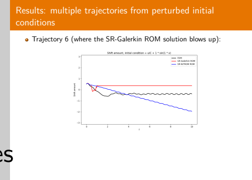
(f) ftd. SR-NiTROM



Results: multiple trajectories from perturbed initial conditions

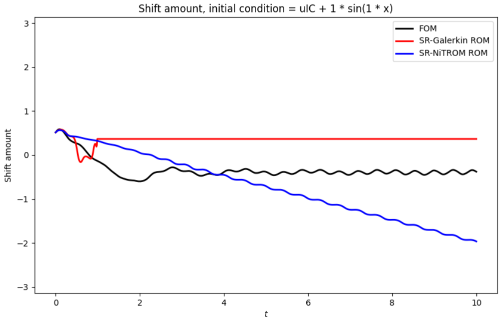
- Trajectory 6 (where the SR-Galerkin ROM solution blows up):





Results: multiple trajectories from perturbed initial conditions

- Trajectory 6 (where the SR-Galerkin ROM solution blows up):

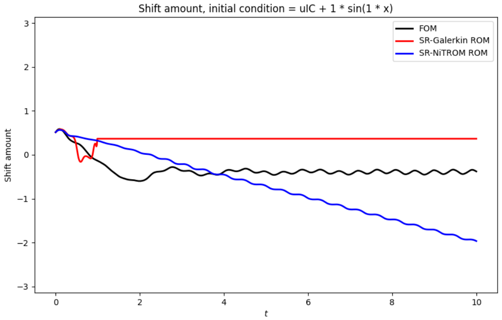


- For this specific trajectory, we find it difficult to train the SR-NiTROM model regardless of fine-tuning the hyperparameters.

Results: multiple trajectories from perturbed initial conditions
Trajectory 6 (where the SR-Galerkin ROM solution blows up):
For this specific trajectory, we find it difficult to train the SR-NiTROM model regardless of fine-tuning the hyperparameters.

Results: multiple trajectories from perturbed initial conditions

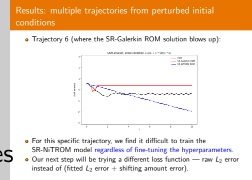
- Trajectory 6 (where the SR-Galerkin ROM solution blows up):



- For this specific trajectory, we find it difficult to train the SR-NiTROM model regardless of fine-tuning the hyperparameters.
- Our next step will be trying a different loss function — raw L_2 error instead of (fitted L_2 error + shifting amount error).

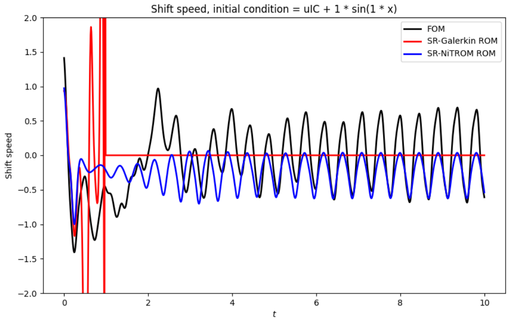
2025-10-29

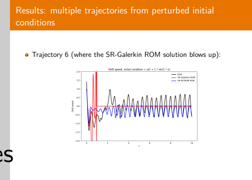
Non-intrusive model reduction of shift-equivariant systems via data-driven projection and reduced dynamics
└Results
└Reconstruction of multiple training KSE trajectories



Results: multiple trajectories from perturbed initial conditions

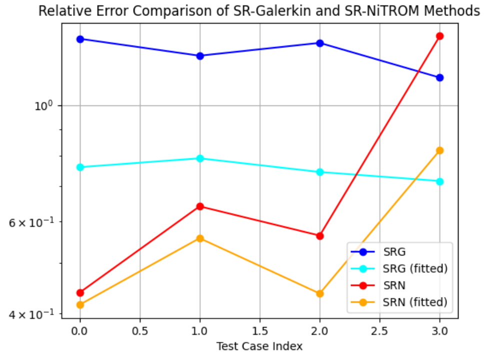
- Trajectory 6 (where the SR-Galerkin ROM solution blows up):





Results: mean reconstruction errors of testing trajectories with 8-dim ROMs

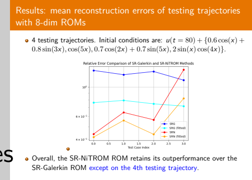
- 4 testing trajectories. Initial conditions are: $u(t=80)+\{0.6\cos(x)+0.8\sin(3x), \cos(5x), 0.7\cos(2x)+0.7\sin(5x), 2\sin(x)\cos(4x)\}$.



- Overall, the SR-NiTROM ROM retains its outperformance over the SR-Galerkin ROM except on the 4th testing trajectory.

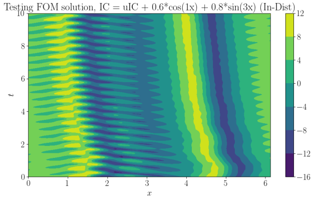
2025-10-29

Non-intrusive model reduction of shift-equivariant systems via data-driven projection and reduced dynamics
└Results
└Reconstruction of multiple testing KSE trajectories

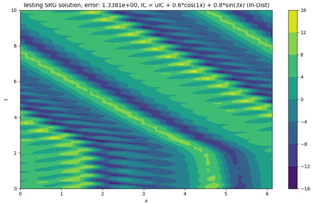


Results: multiple trajectories from perturbed initial conditions

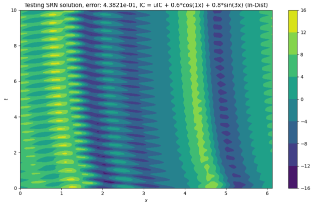
- Testing trajectory 1:



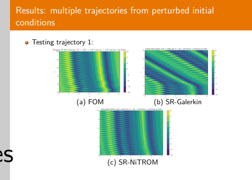
(a) FOM



(b) SR-Galerkin

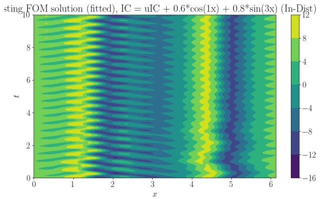


(c) SR-NiTROM

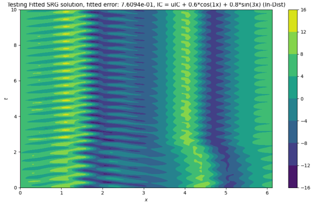


Results: multiple trajectories from perturbed initial conditions

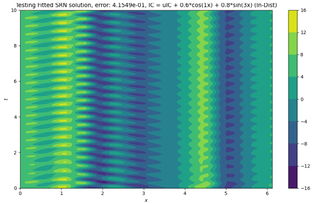
- Testing trajectory 1:



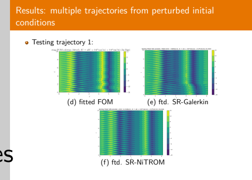
(d) fitted FOM



(e) ftd. SR-Galerkin

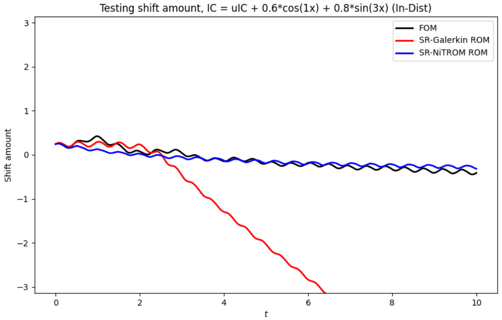


(f) ftd. SR-NiTROM



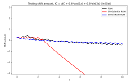
Results: multiple trajectories from perturbed initial conditions

- Testing trajectory 1:



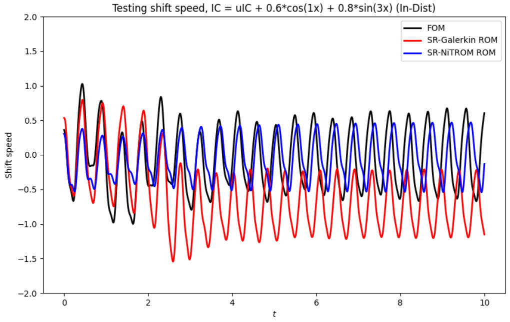
Results: multiple trajectories from perturbed initial conditions

- Testing trajectory 1:



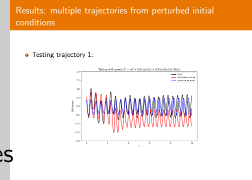
Results: multiple trajectories from perturbed initial conditions

- Testing trajectory 1:



2025-10-29

Non-intrusive model reduction of shift-equivariant systems via data-driven projection and reduced dynamics
└Results
└Reconstruction of multiple testing KSE trajectories



Results: multiple trajectories from perturbed initial conditions

- The same training dataset, 16-dim ROM

Results: multiple trajectories from perturbed initial conditions
- The same training dataset, 16-dim ROM

- The same training dataset, 16-dim ROM

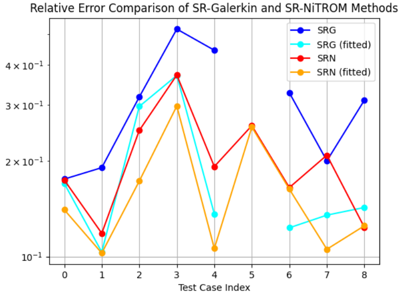
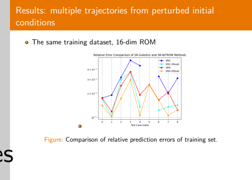


Figure: Comparison of relative prediction errors of training set.

2025-10-29

Non-intrusive model reduction of shift-equivariant systems via data-driven projection and reduced dynamics
└Results
└Reconstruction of multiple training KSE trajectories



Results: multiple trajectories from perturbed initial conditions

- The same training dataset, 16-dim ROM

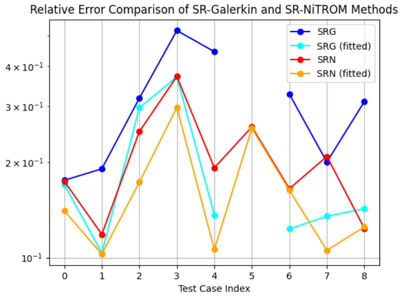


Figure: Comparison of relative prediction errors of training set.

- The SR-NiTROM has lower reconstruction error on average but is outperformed by the SR-Galerkin on the 8th training trajectory.

Results: multiple trajectories from perturbed initial conditions
The same training dataset, 16-dim ROM
Figure: Comparison of relative prediction errors of training set.
The SR-NiTROM has lower reconstruction error on average but is outperformed by the SR-Galerkin on the 8th training trajectory.

Summary: model reduction of KSE system via SR-NiTROM

- With a small set of modes (4 modes), SR-NiTROM has considerably better performance than the SR-Galerkin ROM in reconstructing the transient, spatial profile and the shifting amount of a single trajectory.

Summary: model reduction of KSE system via SR-NiTROM

- With a small set of modes (4 modes), SR-NiTROM has considerably better performance than the SR-Galerkin ROM in reconstructing the transient, spatial profile and the shifting amount of a single trajectory.

Summary: model reduction of KSE system via SR-NiTROM

- With a small set of modes (4 modes), SR-NiTROM has considerably better performance than the SR-Galerkin ROM in reconstructing the transient, spatial profile and the shifting amount of a single trajectory.
- With a moderate number of modes (8 modes), SR-NiTROM has better performance (especially on the shifting amount) and robustness (no blow-up) on average in reconstructing multiple trajectories (based on "curriculum learning").

Summary: model reduction of KSE system via SR-NiTROM

- With a small set of modes (4 modes), SR-NiTROM has considerably better performance than the SR-Galerkin ROM in reconstructing the transient, spatial profile and the shifting amount of a single trajectory.
- With a moderate number of modes (8 modes), SR-NiTROM has better performance (especially on the shifting amount) and robustness (no blow-up) on average in reconstructing multiple trajectories (based on "curriculum learning").

Summary: model reduction of KSE system via SR-NiTROM

- With a small set of modes (4 modes), SR-NiTROM has considerably better performance than the SR-Galerkin ROM in reconstructing the transient, spatial profile and the shifting amount of a single trajectory.
- With a moderate number of modes (8 modes), SR-NiTROM has better performance (especially on the shifting amount) and robustness (no blow-up) on average in reconstructing multiple trajectories (based on "curriculum learning").
- With more number of modes (16 modes), SR-NiTROM gives more reliable reconstruction of the training data ($\approx 10\%$ for every training trajectory), while the SR-Galerkin ROM has higher errors and even blows up sometimes.

Summary: model reduction of KSE system via SR-NiTROM

- With a small set of modes (4 modes), SR-NiTROM has considerably better performance than the SR-Galerkin ROM in reconstructing the transient, spatial profile and the shifting amount of a single trajectory.
- With a moderate number of modes (8 modes), SR-NiTROM has better performance (especially on the shifting amount) and robustness (no blow-up) on average in reconstructing multiple trajectories (based on "curriculum learning").
- With more number of modes (16 modes), SR-NiTROM gives more reliable reconstruction of the training data ($\approx 10\%$ for every training trajectory), while the SR-Galerkin ROM has higher errors and even blows up sometimes.

Summary: model reduction of KSE system via SR-NiTROM

- With a small set of modes (4 modes), SR-NiTROM has considerably better performance than the SR-Galerkin ROM in reconstructing the transient, spatial profile and the shifting amount of a single trajectory.
- With a moderate number of modes (8 modes), SR-NiTROM has better performance (especially on the shifting amount) and robustness (no blow-up) on average in reconstructing multiple trajectories (based on "curriculum learning").
- With more number of modes (16 modes), SR-NiTROM gives more reliable reconstruction of the training data ($\approx 10\%$ for every training trajectory), while the SR-Galerkin ROM has higher errors and even blows up sometimes.
- Further steps: switching the loss function to the raw L_2 error to seek potential improvement.

Summary: model reduction of KSE system via SR-NiTROM

- With a small set of modes (4 modes), SR-NiTROM has considerably better performance than the SR-Galerkin ROM in reconstructing the transient, spatial profile and the shifting amount of a single trajectory.
- With a moderate number of modes (8 modes), SR-NiTROM has better performance (especially on the shifting amount) and robustness (no blow-up) on average in reconstructing multiple trajectories (based on "curriculum learning").
- With more number of modes (16 modes), SR-NiTROM gives more reliable reconstruction of the training data ($\approx 10\%$ for every training trajectory), while the SR-Galerkin ROM has higher errors and even blows up sometimes.
- Further steps: switching the loss function to the raw L_2 error to seek potential improvement.

- Consider the 3D Navier-Stokes equations (3DLNS) linearized about the base flow $\boldsymbol{U} = (U(y), 0, 0)$ for fluctuations (u, v, w, p) of flow field in a channel domain (periodic in streamwise/spanwise direction, no-slip in normal direction):

$$u_t + Uu_x + vU_y = -p_x + \frac{1}{Re}\nabla^2 u \tag{11a}$$

$$v_t + Uv_x = -p_y + \frac{1}{Re}\nabla^2 v \tag{11b}$$

$$w_t + Uw_x = -p_z + \frac{1}{Re}\nabla^2 w \tag{11c}$$

$$u(y = \pm 1) = v(y = \pm 1) = w(y = \pm 1) = 0 \tag{11d}$$

- Why we want to study this full-order model?
  - The solutions to 3DLNS has large transient growth before the final decay (e.g., the Orr mechanism, the lift-up mechanism, streamwise elongated structures).
  - In 3D Navier-Stokes system, the transient will trigger nonlinear effects before decay, resulting in transition to turbulence.

Numerical details

- Change of variables: denote the normal vorticity $\eta = u_z - w_x$ and eliminate p (substitute $\nabla^2 p$ into (11b)), we can rewrite the 3DLNS using $q = (v, \eta)$ as state variables.

$$q_t = Lq \tag{12a}$$

$$L = \begin{bmatrix} \nabla^2 & \\ & I \end{bmatrix}^{-1} \begin{bmatrix} -U\partial_x \nabla^2 + U_{yy}\partial_x + \nabla^4/\mathrm{Re} & \\ -U_y \partial_z & -U\partial_x + \nabla^2/\mathrm{Re} \end{bmatrix} \tag{12b}$$

$$v(y = \pm 1) = v_y(y = \pm 1) = \eta(y = \pm 1) = 0. \tag{12c}$$

- Spatial discretization:
  - Fourier-based differentiation in x and z direction.
  - Chebyshev collocation method in y direction.
  - Enforcing the clamped boundary conditions.
- Time stepping: fourth-order Runge-Kutta exponential time differencing scheme (ETDRK4, Cox et al. 2002)
  - Equal to the exponential time stepping $q(t + \Delta t) = e^{\Delta t L} q(t)$ without external forcing.

Numerical details

- Re $= 3000$. Initial conditions (Henningson et al. 1993):

$$\psi = (1 - y^2)^2 (x/2) z e^{-(x/2)^2 - (z/2)^2} \tag{13}$$

$$(u, v, w) = (0, \psi_z, -\psi_y) \tag{14}$$

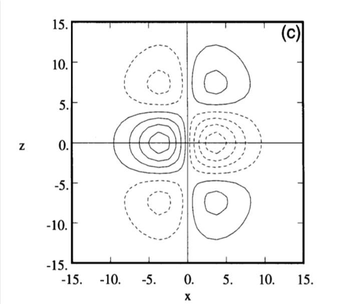
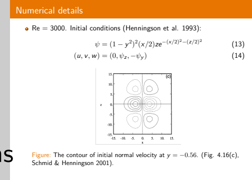
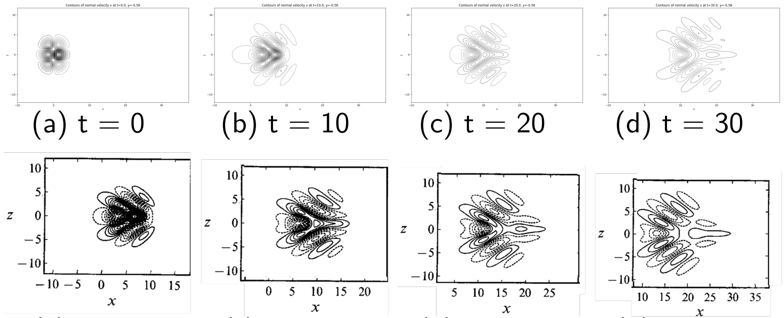


Figure: The contour of initial normal velocity at $y = -0.56$. (Fig. 4.16(c), Schmid & Henningson 2001).



Results: simulation of 3DLNS compared with previous studies



(a) t = 0 (b) t = 10 (c) t = 20 (d) t = 30

(e) t = 0 (f) t = 10 (g) t = 20 (h) t = 30

Figure: The contour of normal velocity at $y = -0.56$ and $t = 0, 10, 20, 30$ compared to Fig. 2 in Henningson et al. (1993).